DFS Application: Topological Sort & Strongly Connected Components

Week-10, Lecture-02

Course Code: CSE221

Course Title: Algorithms

Program: B.Sc. in CSE

Course Teacher: Tanzina Afroz Rimi

Designation: Lecturer

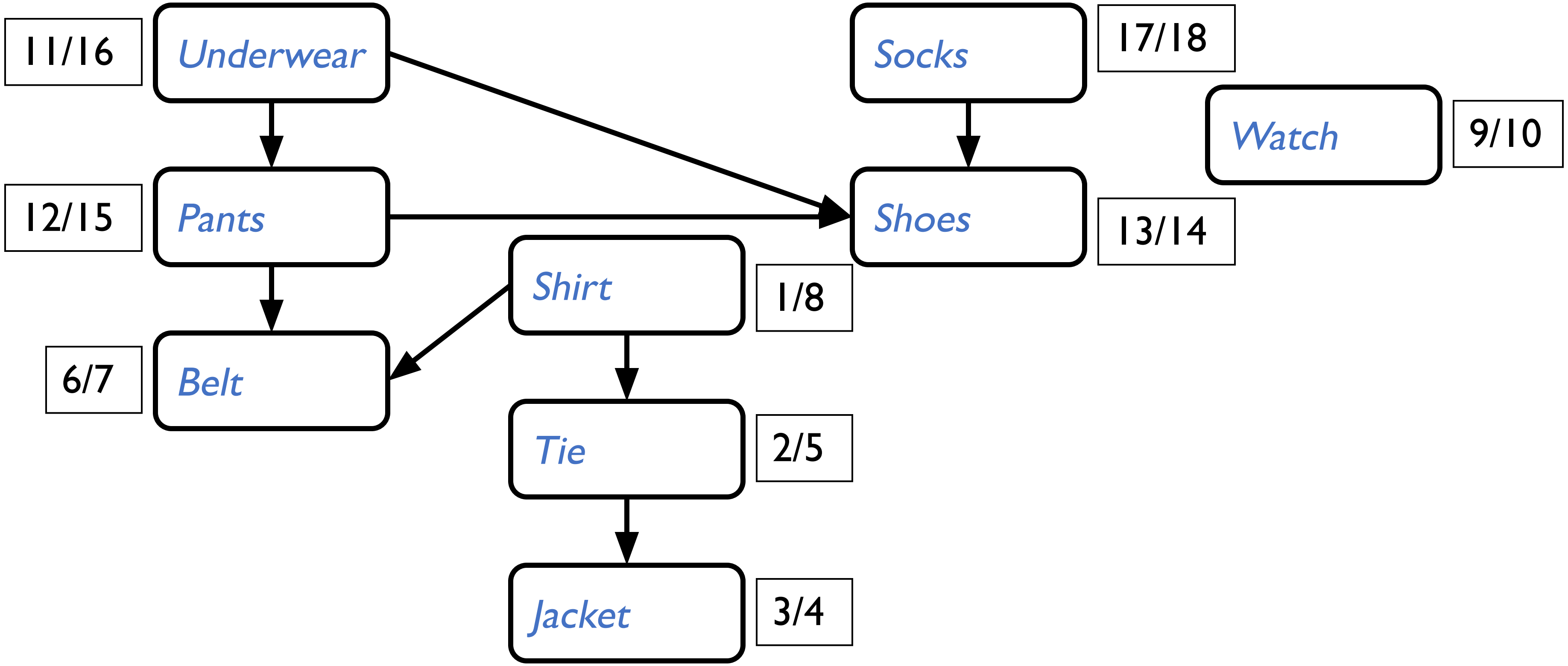
Email: tanzinaafroz.cse@diu.edu.bd

Topological Sort

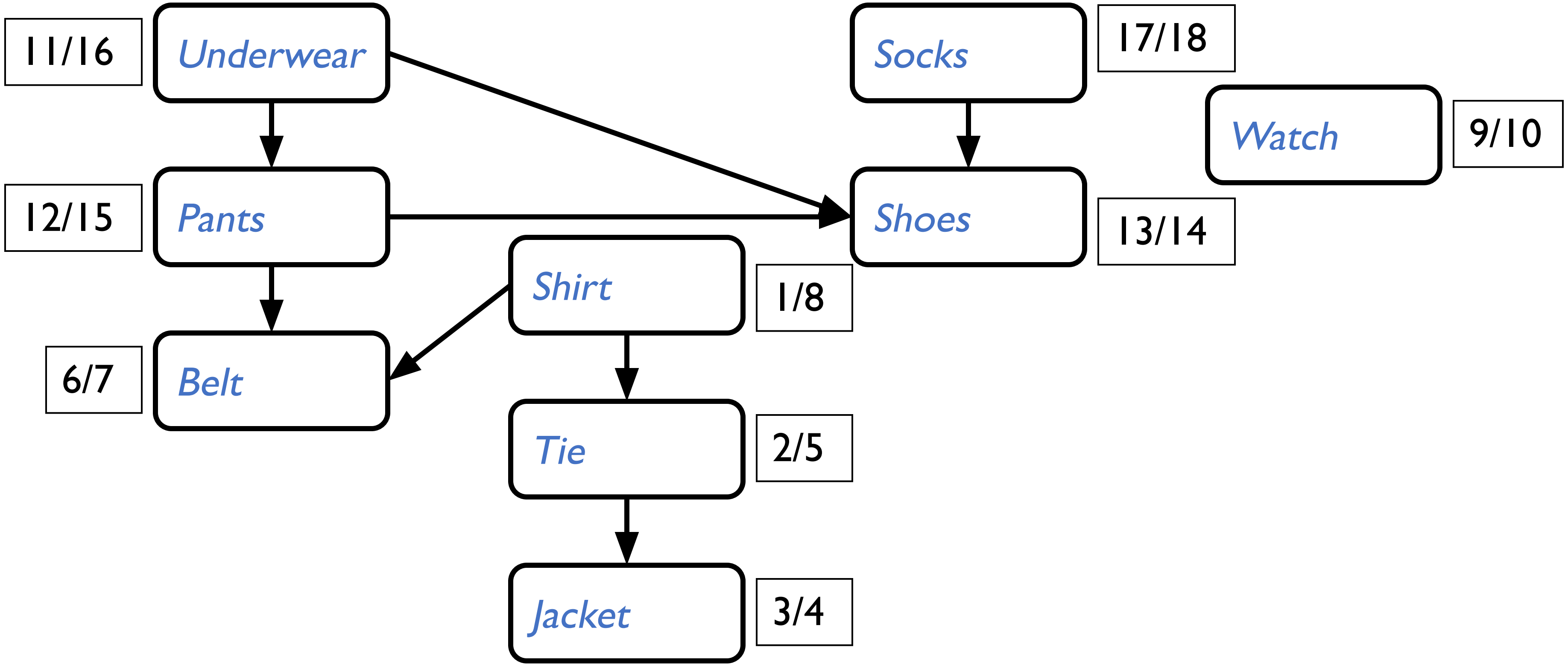
Topological Sort

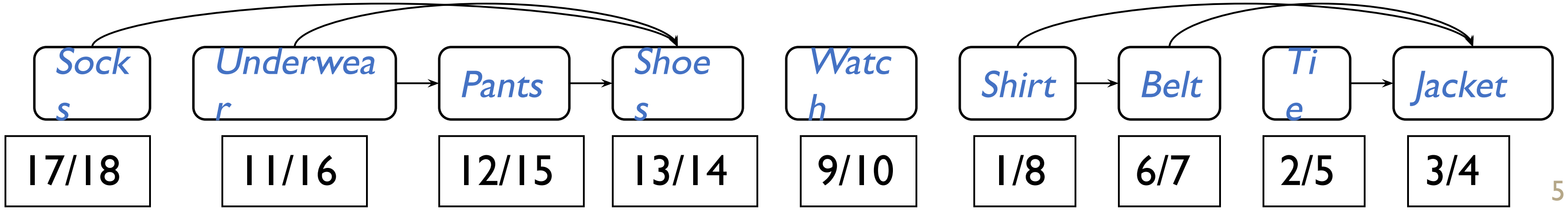
- *Topological sort* of a DAG:
  - Linear ordering of all vertices in graph G such that vertex u comes before vertex v if edge $(u, v) \in G$
- Real-world example: getting dressed

Getting Dressed



Getting Dressed





Topological Sort Algorithm

```
Topological-Sort()
{
   Run DFS
   When a vertex is finished, output it
   Vertices are output in reverse topological order
}
```

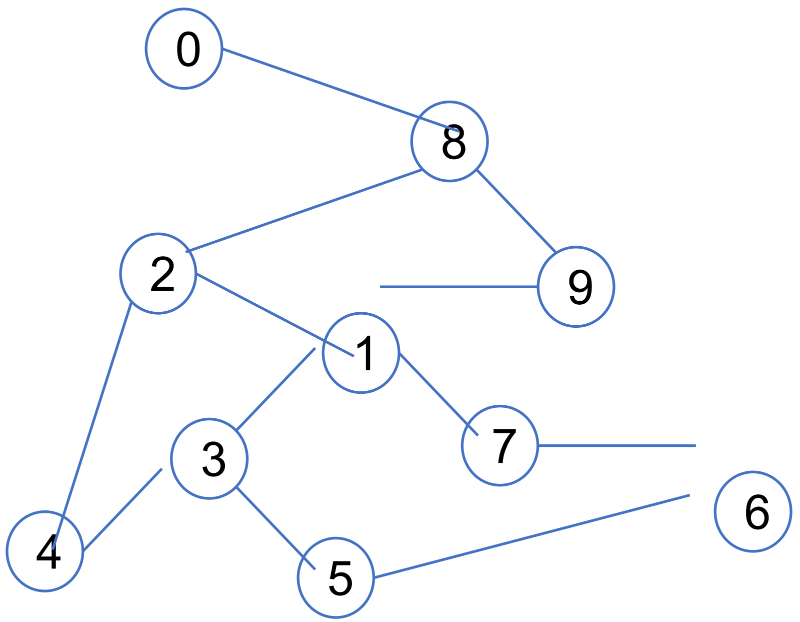
- Time: O(V+E)
- Correctness: Want to prove that
  $(u,v) \in G \Rightarrow u\rightarrow f > v\rightarrow f$

Strongly Connected Component (SCC)

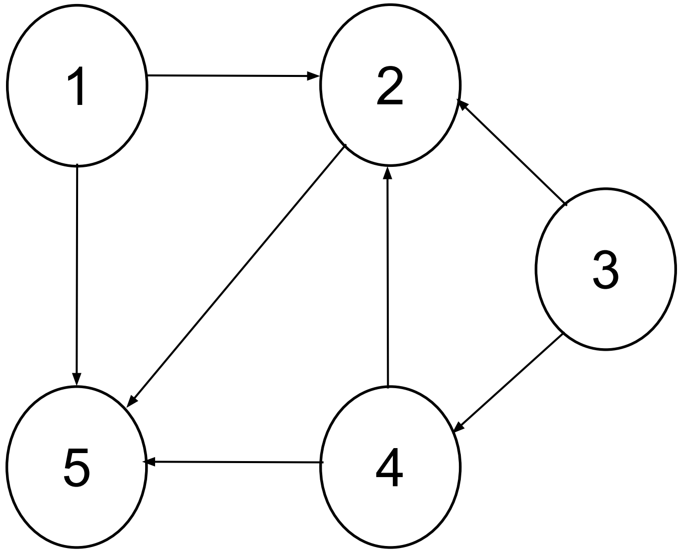
Connectivity

- Connected Graph
  - In an **<u>undirected graph</u>** G, two vertices u and v are called connected if G contains a path from u to v. Otherwise, they are called disconnected.
  - A **<u>directed graph</u>** is called connected if every pair of distinct vertices in the graph is connected.
- Connected Components
  - A connected component is a maximal connected subgraph of G. Each vertex belongs to exactly one connected component, as does each edge.

Connectivity (cont.)



Connected Undirected Graph



Connected Directed Graph

Connectivity (cont.)

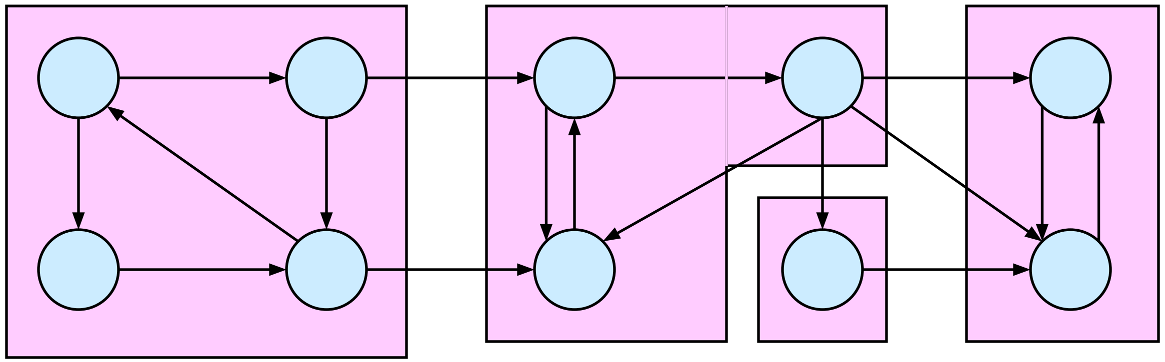
- Weakly Connected Graph
  - A directed graph is called **weakly connected** if replacing all of its directed edges with undirected edges produces a connected (undirected) graph.
- Strongly Connected Graph
  - It is strongly connected or strong if it contains a directed path from u to v for every pair of vertices u, v. The strong components are the maximal strongly connected subgraphs

Connected Components

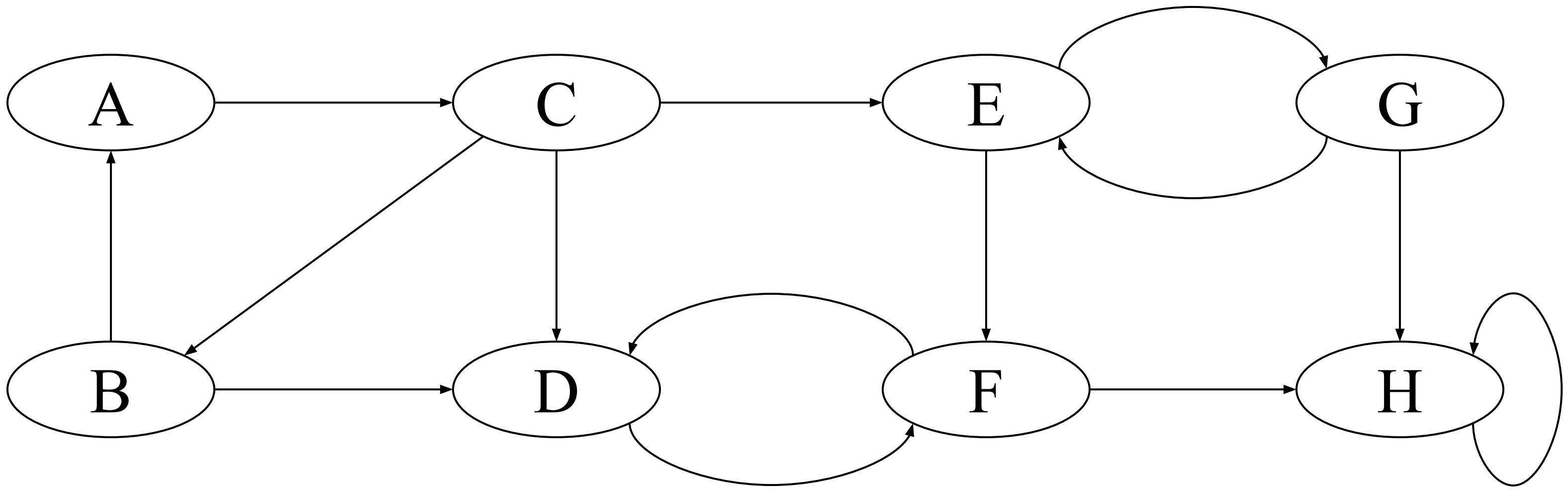
- Strongly connected graph
  - A directed graph is called *strongly connected* if for every pair of vertices *u* and *v* there is a path from *u* to *v* and a path from *v* to *u*.
- Strongly Connected Components (SCC)
  - The **strongly connected components** (**SCC**) of a directed graph are its maximal strongly connected sub-graphs.
- Here, we work with
  - Directed unweighted graph

Strongly Connected Components

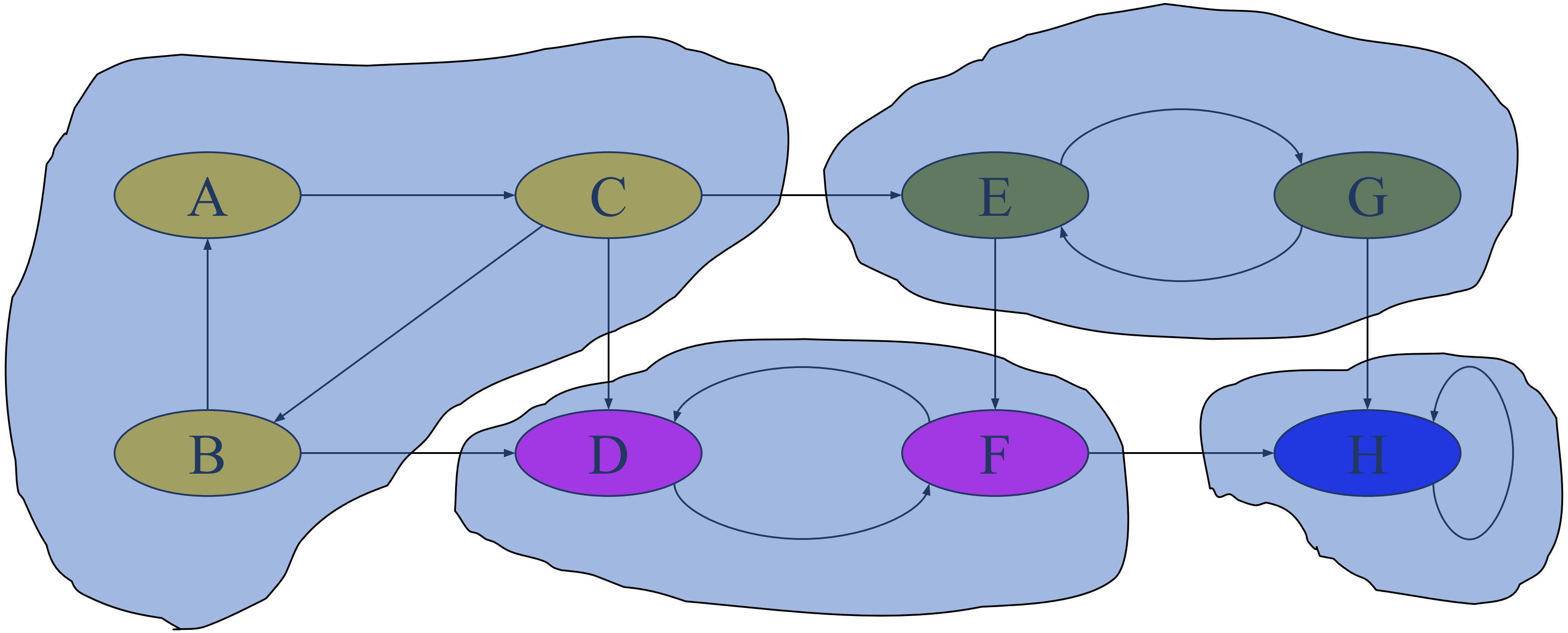
- G is strongly connected if every pair (u, v) of vertices in G is reachable from one another.
- A **strongly connected component** (*SCC*) of G is a maximal set of vertices $C \subseteq V$ such that for all $u, v \in C$, both $u \rightsquigarrow v$ and $v \rightsquigarrow u$ exist.



Strongly Connected Components

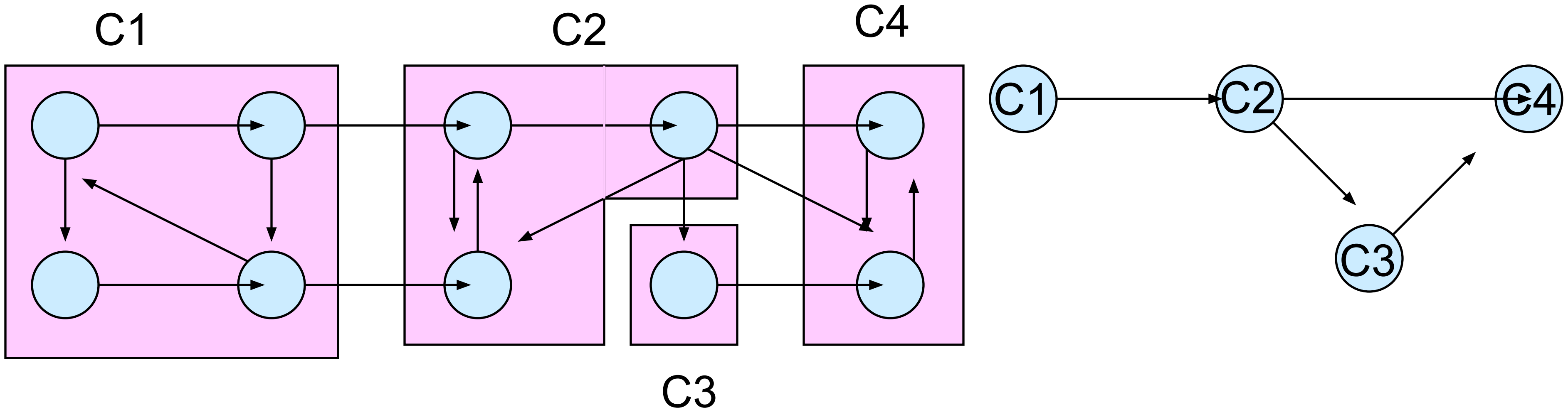


Strongly Connected Components



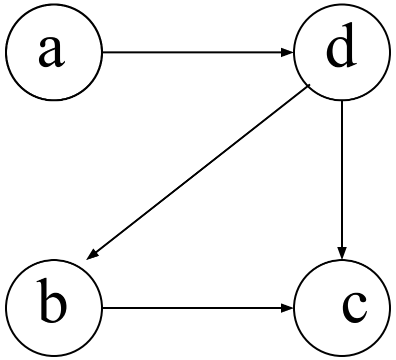
Component Graph

- $G^{SCC} = (V^{SCC}, E^{SCC})$.
- V^{SCC} has one vertex for each SCC in G.
- E^{SCC} has an edge if there's an edge between the corresponding SCC's in G.
- G^{SCC} for the example considered:

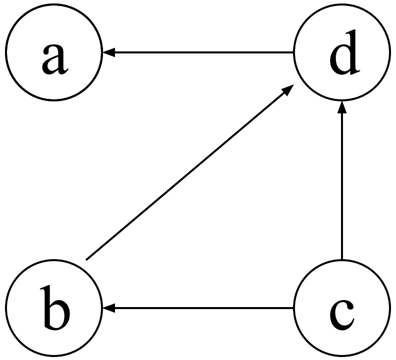


Strongly Connected Components

The **transpose** M^T of an NxN matrix M is the matrix obtained when the rows become columns and the column become rows:



G



$\mathbf{G^T}$

Edges have reverse direction!

M

	a	b	c	d
a				1
b			1	
c				
d		1	1	

$\mathbf{M^T}$

	a	b	c	d
a				
b				1
c		1		1
d	1			

Transpose of a Directed Graph

- G^{T} = **transpose** of directed G.
  - $G^{\mathrm{T}} = (V, E^{\mathrm{T}})$, $E^{\mathrm{T}} = \{(u, v) : (v, u) \in E\}$.
  - G^{T} is G with all edges reversed.
- Can create G^{T} in $\Theta(V + E)$ time if using adjacency lists.
- G and G^{T} have the *same* SCC's. (u and v are reachable from each other in G if and only if reachable from each other in G^{T}.)

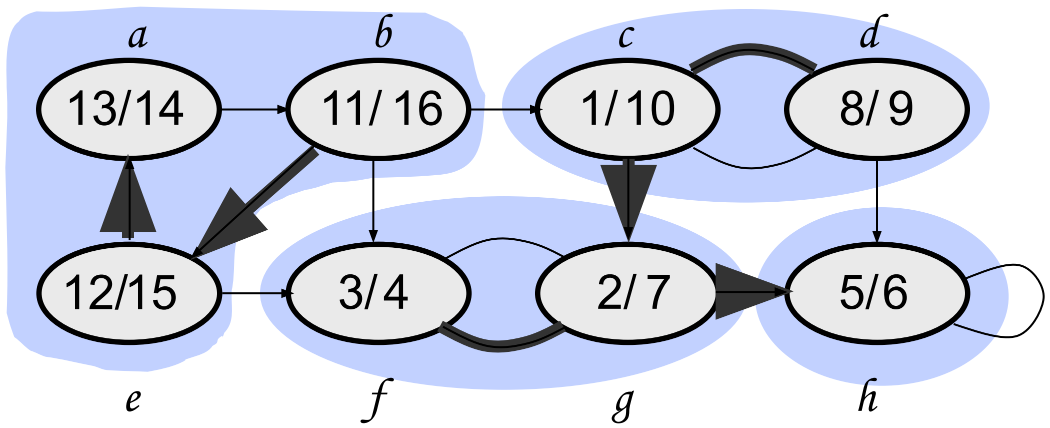
Algorithm to determine SCCs

SCC(G)

1. call DFS(G) to compute finishing times $f[u]$ for all u
2. compute G^T
3. call DFS(G^T), but in the main loop, consider vertices in order of decreasing $f[u]$ (as computed in first DFS)
4. output the vertices in each tree of the depth-first forest formed in second DFS as a separate SCC

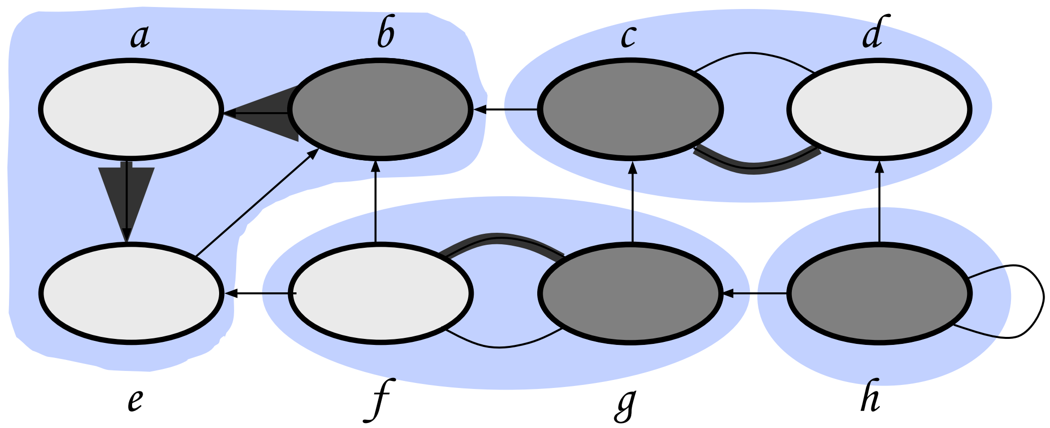
Time: $\Theta(V + E)$.

Example



DFS on the initial graph G

b	e	a	c	d	g	h	f
16	15	14	10	9	7	6	4

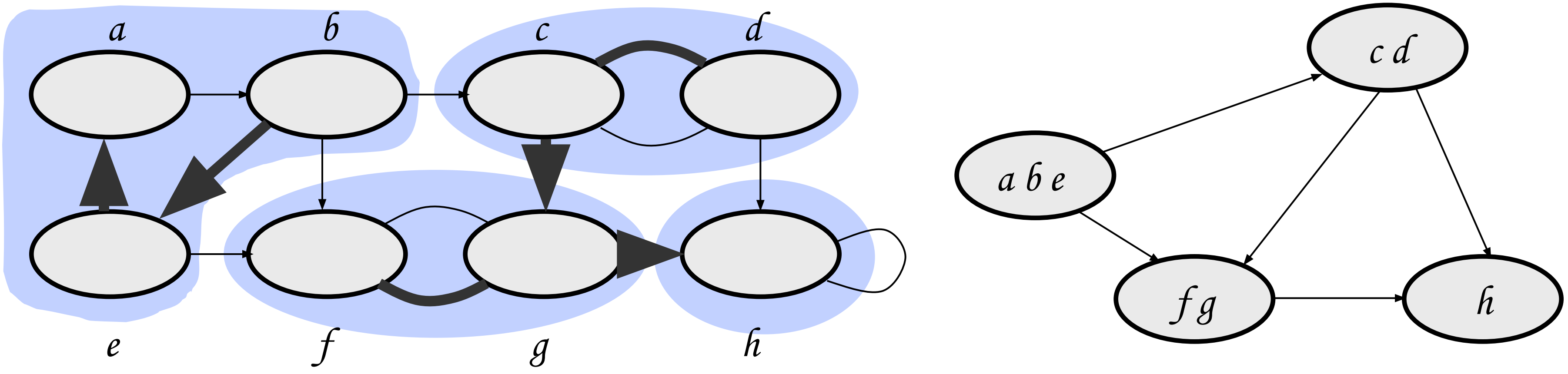


DFS on G^T:

- start at b: visit a, e
- start at c: visit d
- start at g: visit f
- start at h

Strongly connected components: C_1 = {a, b, e}, C_2 = {c, d}, C_3 = {f, g}, C_4 = {h}

Component Graph



- The **component graph** $G^{SCC} = (V^{SCC}, E^{SCC})$:
  - $V^{SCC} = \{v_1, v_2, \ldots, v_k\}$, where v_i corresponds to each strongly connected component C_i
  - There is an edge $(v_i, v_j) \in E^{SCC}$ if G contains a directed edge (x, y) for some $x \in C_i$ and $y \in C_j$
- The component graph is a DAG

Textbooks & Web References

- Text Book (Chapter 22)
- www.geeksforgeeks.org

Thank you

&

Any question?